

INEXACT OBJECTIVE FUNCTION EVALUATIONS IN A TRUST-REGION ALGORITHM FOR PDE-CONSTRAINED OPTIMIZATION UNDER UNCERTAINTY

D. P. KOURI*, M. HEINKENSCHLOSS†, D. RIDZAL‡, AND B. G. VAN BLOEMEN
WAANDERS§

Abstract. This paper improves the trust-region algorithm with adaptive sparse grids introduced in [?] for the solution of optimization problems governed by partial differential equations (PDEs) with uncertain coefficients. The previous algorithm used adaptive sparse grid discretizations to generate models that are applied in a trust-region framework to generate a trial step. The decision whether to accept this trial step as the new iterate, however, required relatively high fidelity adaptive discretizations of the objective function. In this paper, we extend the algorithm and convergence theory to allow the use of low-fidelity adaptive sparse-grid models in objective function evaluations. This is accomplished by extending conditions on inexact function evaluations used in previous trust-region frameworks. Our algorithm adaptively builds two separate sparse grids: one to generate optimization models for the step computation, and one to approximate the objective function. These adapted sparse grids typically contain significantly fewer points than the high-fidelity grids, which leads to a dramatic reduction in the computational cost. This is demonstrated numerically using two examples. Moreover, the numerical results indicate that the new algorithm rapidly identifies the stochastic variables that are relevant to obtaining an accurate optimal solution. When the number of such variables is independent of the dimension of the stochastic space, the algorithm exhibits near dimension-independent behavior.

Key words. PDE-constrained optimization, uncertainty, stochastic collocation, trust regions, sparse grids, adaptivity

AMS subject classifications. 49M15, 65K05, 65N35, 90C15

1. Introduction. The solution of large-scale optimization problems in science and engineering must accommodate model uncertainties, such as unknown material properties and boundary conditions. The coupling of traditional optimization methods with uncertainty quantification faces significant computational challenges due to the potentially large number of stochastic variables. To address this issue, we have developed an algorithm that shows promising results for optimization problems governed by partial differential equations (PDEs) with random coefficients [?]. This algorithm uses a trust-region framework to manage models that are based on sparse grids. In [?] we use adaptive sparse grids to compute the optimization step and a fixed high-fidelity sparse grid to determine whether to accept the step. As discussed in [?], the dominant

*Optimization and Uncertainty Quantification, MS-1320, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185-1320. E-mail: dpkouri@sandia.gov. Research sponsored by the U.S. DOE Office of Science J. H. Wilkinson Fellowship and the NNSA Advanced Scientific Computing (ASC) Program.

†Department of Computational and Applied Mathematics, MS-134, Rice University, 6100 Main Street, Houston, TX 77005-1892. E-mail: heinken@rice.edu. This author was supported in part by AFOSR grant FA9550-12-1-0155, and by NSF grant DMS-1115345.

‡Optimization and Uncertainty Quantification, MS-1320, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185-1320. E-mail: dridzal@sandia.gov. Research sponsored by the NNSA Advanced Scientific Computing (ASC) Program.

§Optimization and Uncertainty Quantification, MS-1320, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185-1320. E-mail: bartv@sandia.gov. Research sponsored by the NNSA Advanced Scientific Computing (ASC) Program. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

and potentially prohibitive computational cost of the algorithm is the evaluation of the objective function using the high-fidelity sparse grid. In this paper we improve the algorithm by also allowing sparse-grid adaptivity in the approximation of the objective function. We prove first-order convergence of the algorithm and demonstrate that the computational cost is reduced dramatically due to our improvement.

The main algorithmic improvements are an extension of the conditions for inexact function evaluations in trust-region methods [?, ?, ?] and the integration with the adaptive sparse-grid discretizations [?] to enforce these conditions in our application context. Our extension of the conditions on inexact function evaluations is applicable to a broad class of problems. The practical difficulty with the conditions on inexact function evaluations presented in [?, ?], [?, Sec. 10.6] is that the error in function values must be bounded by a specific value related to the trust-region algorithm. Such conditions are impossible to enforce through error estimates that involve unknown constants. Our modification of these conditions is motivated by [?, ?], where a trust-region framework and sequential quadratic programming methods are applied to determine adaptive finite-element discretizations for PDE-constrained optimization. For the convergence result in [?] it is assumed that if infinitely many objective function approximations are generated, the error estimator converges to zero. This is not guaranteed in our application. Therefore, we introduce a forcing sequence. Moreover, our conditions are more directly tied to those in [?, ?], [?, Sec. 10.6], which in our setting are satisfied asymptotically.

As in [?], the inexact gradient considerations are adapted from [?]. We consider the classic trust-region framework [?] and the retrospective trust-region framework of [?]. The classic uses the current model to accept a step and to update the trust-region radius. The retrospective also uses the current model to accept a step, but employs a new model to update the trust-region radius. This may lead to larger radii and faster convergence in practice.

Our new, fully adaptive strategy builds two separate sparse grids: one to model the derivative information that is used to define trust-region subproblems and one to approximate the objective function. In practice, the adaptively built sparse grids often contain significantly fewer points than high-fidelity grids.

This paper is organized as follows. In Sections 2 and 3 we summarize our problem formulation, the sparse-grid collocation discretization, and the notation needed to analyze our algorithmic improvements. The content of these sections is discussed in greater detail in [?]. Section 4 presents the new fully adaptive trust-region algorithm for PDE-constrained optimization under uncertainty using both the classic and the retrospective version of the trust-region method. Convergence to a point satisfying the first-order optimality conditions is proved in Appendix A. Numerical results presented in Section 5 show a vast reduction in the number of PDE solves required to compute an optimal solution, when compared with [?].

2. Problem Formulation. Let (Ω, \mathcal{F}, P) denote a complete probability space, where Ω is the set of outcomes, $\mathcal{F} \subseteq 2^\Omega$ is a σ -algebra of events, and $P : \mathcal{F} \rightarrow [0, 1]$ is a probability measure. Furthermore let \mathcal{V} and \mathcal{Z} be real Hilbert spaces of deterministic functions defined on the physical domain $D \subset \mathbb{R}^d$, $d = 1, 2, 3$, and let $\hat{\mathbf{A}} : \Omega \rightarrow \mathcal{L}(\mathcal{V}, \mathcal{V}^*)$, $\hat{\mathbf{B}} : \Omega \rightarrow \mathcal{L}(\mathcal{Z}, \mathcal{V}^*)$, $\hat{\mathbf{b}} : \Omega \rightarrow \mathcal{V}^*$, and $\hat{\mathbf{N}} : \mathcal{V} \times \Omega \rightarrow \mathcal{V}^*$. We consider nonlinear PDEs of the form

$$\hat{\mathbf{A}}(\omega)u(\omega) + \hat{\mathbf{N}}(u(\omega), \omega) + \hat{\mathbf{B}}(\omega)z + \hat{\mathbf{b}}(\omega) = 0 \quad \text{a.e. in } \Omega.$$

To facilitate computation, we employ the finite-dimensional noise assumption. That is, we assume there exists an M -dimensional random vector $Y = (Y_1, \dots, Y_M) : \Omega \rightarrow \Gamma$ with $Y_k : \Omega \rightarrow \Gamma_k \subseteq \mathbb{R}$ and $\Gamma = \Gamma_1 \times \dots \times \Gamma_M$ such that $\widehat{\mathbf{A}}(\omega) \equiv \mathbf{A}(Y(\omega))$, $\widehat{\mathbf{B}}(\omega) \equiv \mathbf{B}(Y(\omega))$, $\widehat{\mathbf{b}}(\omega) \equiv \mathbf{b}(Y(\omega))$, and $\widehat{\mathbf{N}}(\cdot, \omega) \equiv \mathbf{N}(\cdot, Y(\omega))$ for some $\mathbf{A} : \Gamma \rightarrow \mathcal{L}(\mathcal{V}, \mathcal{V}^*)$, $\mathbf{B} : \Gamma \rightarrow \mathcal{L}(\mathcal{Z}, \mathcal{V}^*)$, $\mathbf{b} : \Gamma \rightarrow \mathcal{V}^*$, and $\mathbf{N} : \mathcal{V} \times \Gamma \rightarrow \mathcal{V}^*$. Moreover, Y is endowed with the joint Lebesgue density $\rho = \rho_1 \otimes \dots \otimes \rho_M$ with $\rho_k : \Gamma_k \rightarrow [0, +\infty) \cup \{+\infty\}$. This assumption permits the change of variables

$$\mathbf{A}(y)u(y) + \mathbf{N}(u(y), y) + \mathbf{B}(y)z + \mathbf{b}(y) = 0, \quad y \in \Gamma. \quad (2.1)$$

Throughout, $u \in L^2_\rho(\Gamma; \mathcal{V})$ is the random-field state variable, and $z \in \mathcal{Z}$ is the deterministic control variable.

We consider optimization problems governed by the PDE (2.1). Let \mathcal{W} denote a real Hilbert space, $\mathbf{C} \in \mathcal{L}(\mathcal{V}, \mathcal{W})$, $\bar{w} \in \mathcal{W}$, and $\alpha > 0$. Consider the optimization problem

$$\min_{z \in \mathcal{Z}} J(z) \stackrel{\text{def}}{=} \frac{1}{2} \mathbb{E} \left[\|\mathbf{C}u(\cdot; z) - \bar{w}\|_{\mathcal{W}}^2 \right] + \frac{\alpha}{2} \|z\|_{\mathcal{Z}}^2, \quad (2.2)$$

where $\mathbb{E}[X] = \int_\Gamma \rho(y) X(y) dy$ denotes the expected value operator and $u(y; z) = u(y) \in \mathcal{V}$ solves (2.1) for almost all $y \in \Gamma$. Under Assumptions 2.2, 2.3, and 2.5 of [?], the gradient of the objective function $J(z)$ in (2.2) is

$$\nabla J(z) = \alpha z + \int_\Gamma \rho(y) \mathbf{B}(y)^* p(y) dy = \alpha z + \mathbb{E}[\mathbf{B}^* p] \quad (2.3)$$

where $p(y; z) = p(y) \in \mathcal{V}$ solves the adjoint equation

$$\mathbf{A}(y)^* p(y) + \mathbf{N}_v(u(y), y)^* p(y) = -\mathbf{C}^*(\mathbf{C}u(y; z) - \bar{w}), \quad y \in \Gamma \quad (2.4)$$

and \mathbf{N}_v denotes the Fréchet derivative of \mathbf{N} with respect to the \mathcal{V} (state) component.

3. Stochastic Collocation. As in [?], we (semi-)discretize the optimal control problem (2.2) by replacing the expected value \mathbb{E} with a sparse grid quadrature approximation. We briefly review this approach to introduce the notation and basic concepts needed for the remainder of the paper.

3.1. Stochastic Collocation for Optimization. Given an appropriate quadrature operator

$$\mathbb{E}_Q[X] = \sum_{k=1}^Q \omega_k X(y_k) \approx \mathbb{E}[X]$$

where $\{(\omega_k, y_k)\}_{k=1}^Q \subset \mathbb{R} \times \Gamma$ denote the quadrature weights and points, we discretize the objective function in (2.2) as

$$J_Q(z) \stackrel{\text{def}}{=} \frac{1}{2} \sum_{k=1}^Q \omega_k \|\mathbf{C}u_k(z) - \bar{w}\|_{\mathcal{W}}^2 + \frac{\alpha}{2} \|z\|_{\mathcal{Z}}^2, \quad (3.1)$$

where $u_k(z) = u_k \in \mathcal{V}$ is the solution of

$$\mathbf{A}(y_k)u_k + \mathbf{N}(u_k, y_k) + \mathbf{B}(y_k)z + \mathbf{b}(y_k) = 0, \quad k = 1, \dots, Q. \quad (3.2)$$

This approach results in the semi-discretized optimization problem

$$\min_{z \in \mathcal{Z}} J_Q(z). \quad (3.3)$$

Again, Assumptions 2.2, 2.3, and 2.5 of [?] imply that the objective function in (3.3) is Fréchet differentiable with gradient

$$\nabla J_Q(z) = \alpha z + \sum_{k=1}^Q \omega_k \mathbf{B}_k^* p_k, \quad (3.4)$$

where $p_k \in \mathcal{V}$ solves the adjoint equations

$$\mathbf{A}(y_k)^* p_k + \mathbf{N}_v(u_k, y_k)^* p_k = -\mathbf{C}^*(\mathbf{C}u_k - \bar{w}), \quad k = 1, \dots, Q. \quad (3.5)$$

As pointed out in [?, ?], sparse grids often result in negative quadrature weights ω_k . In such cases, it is not clear that the semi-discretized problem (3.3) is well-posed.

3.2. Sparse Grids. The material in this section is based on previous work by many authors, including [?, ?, ?, ?, ?].

Sparse-grid quadrature operators are constructed from one-dimensional (1D) quadrature operators. For $k = 1, \dots, M$, let $\{\mathbb{E}_k^i\}_{i \geq 1}$ denote a sequence of 1D quadrature operators built on the quadrature points $\mathcal{N}_k^i \subset \Gamma_k$ such that \mathbb{E}_k^i is exact for polynomials of $d_k^i - 1$, where $\{d_k^i\}_{i=1}^\infty \subset \mathbb{N}$ is an increasing sequence, and

$$\mathbb{E}_k^i[X] \rightarrow \mathbb{E}_k[X] = \int_{\Gamma_k} \rho_k(y) X(y) dy \quad \text{as } i \rightarrow \infty$$

for sufficiently regular $X \in C_{\rho_k}^0(\Gamma_k)$. Define the 1D difference quadrature operators

$$\Delta_k^1 \stackrel{\text{def}}{=} \mathbb{E}_k^1 \quad \text{and} \quad \Delta_k^i \stackrel{\text{def}}{=} \mathbb{E}_k^i - \mathbb{E}_k^{i-1}, \quad \text{for } i \geq 2.$$

To define the M -D quadrature rule on $\Gamma = \Gamma_1 \times \dots \times \Gamma_M$ let $\mathbf{i} = (i_1, \dots, i_M)$ be a multi-index and let $\mathcal{I} \subset \mathbb{N}_+^M$ be a finite multi-index set, where $\mathbb{N}_+ = \{1, 2, \dots\}$. The general sparse-grid quadrature operator is defined as

$$\mathbb{E}_{\mathcal{I}} \stackrel{\text{def}}{=} \sum_{\mathbf{i} \in \mathcal{I}} (\Delta_1^{i_1} \otimes \dots \otimes \Delta_M^{i_M}). \quad (3.6)$$

The quadrature rule in (3.6) is expressed via 1D difference quadrature operators Δ_k^i . If the index set $\mathcal{I} \subset \mathbb{N}_+^M$ is admissible, in the sense that for all $\mathbf{i} = (i_1, \dots, i_M) \in \mathcal{I}$ it holds that

$$\mathbf{j} = (j_1, \dots, j_M) \in \mathbb{N}_+^M \quad \text{and} \quad j_k \leq i_k \quad \forall k = 1, \dots, M \quad \implies \quad \mathbf{j} \in \mathcal{I},$$

then we can use the combination technique [?] to write (3.6) in terms of the original 1D quadrature operators \mathbb{E}_k^i . Furthermore, we can determine the set of points required to evaluate $\mathbb{E}_{\mathcal{I}}$ (i.e., the sparse grid associated with \mathcal{I}) and the sparse grid collocation weights can be computed from the weights of the original 1D quadrature formulas. See, e.g., [?] or [?] for details.

To determine the admissible index set in (3.6) we use the dimension-adaptive approach presented in [?]. The application of this approach in our context is presented in Sections 4.2 and 4.3.

Notation. To emphasize the dependence of the objective function approximation on the sparse grid, we will use the notation $J_{\mathcal{I}}(z)$ and $\mathbb{E}_{\mathcal{I}}$ instead of $J_Q(z)$ and \mathbb{E}_Q .

4. Trust Regions. To facilitate numerical computation, we wish to approximate J and its gradient using adaptive sparse grids. Given a current iterate z_k , trust-region methods compute a new iterate by approximately minimizing a local model $m_k(s)$ of the true objective function $J(z_k + s)$ over the trust region $\{s \in \mathcal{Z} : \|s\| \leq \Delta_k\}$. As in [?] we use sparse-grid models $m_k(s) = J_{\mathcal{I}_k}(z_k + s)$. To guarantee convergence of the trust-region method, the gradient of the model, $\nabla m_k(0) = \nabla J_{\mathcal{I}_k}(z_k)$, must be a sufficiently good approximation of the true gradient, $\nabla J(z_k)$. We specify the gradient approximation quality in (4.3) below and discuss how to enforce this condition in Section 4.2.

Once an approximate minimizer s_k of the model m_k is computed, trust-region algorithms check whether $z_k + s_k$ is accepted as the new iterate, and the trust-region radius is updated. Step acceptance and the trust-region update depend on the ratio of actual reduction $\text{ared}_k = (J(z_k) - J(z_k + s_k))$ and predicted reduction $\text{pred}_k = (m_k(0) - m_k(s_k))$. The actual reduction involves the exact objective function J in (2.2), which we cannot compute. In [?] we used a fixed high-fidelity sparse grid with index set \mathcal{J}_F and replaced ared_k by $J_{\mathcal{J}_F}(z_k) - J_{\mathcal{J}_F}(z_k + s_k)$. In this paper we replace \mathcal{J}_F with an adaptive index set \mathcal{J}_k , and use $J_{\mathcal{J}_k}(z_k) - J_{\mathcal{J}_k}(z_k + s_k)$ instead of ared_k .

Prior references [?, ?, ?] and [?, Secs. 8.4, 10.6] postulate conditions on the gradient and objective function accuracy that are impossible to verify in many applications. In the next section, we extend the work of [?, ?] to relax these conditions for the classic and retrospective trust-region algorithms. In Sections 4.2 and 4.3 we discuss the implementation of our inexact gradient and objective function conditions for (2.2) using adaptive sparse grids. The proposed algorithms apply to not only (2.2), but also infinite-dimensional optimization problems where objective functions are approximated by adaptive finite elements [?, ?], or reduced-order models [?, ?].

4.1. The Algorithm. Let $J : \mathcal{Z} \rightarrow \mathbb{R}$ be a smooth functional on a Hilbert space. Precise assumptions on the smoothness of J are stated below. Given an iterate z_k , a trust-region algorithm builds a smooth model $m_k : \mathcal{Z} \rightarrow \mathbb{R}$ of $s \mapsto J(z_k + s)$ on the trust-region $\{s \in \mathcal{Z} : \|s\| \leq \Delta_k\}$, where $\Delta_k > 0$ is the trust-region radius. The algorithm then computes a trial step s_k by approximately solving

$$\min_{s \in \mathcal{Z}} m_k(s) \quad \text{subject to} \quad \|s\|_{\mathcal{Z}} \leq \Delta_k. \quad (4.1)$$

The trial step s_k must satisfy the fraction of Cauchy decrease condition

$$m_k(0) - m_k(s_k) \geq \kappa_0 \|\nabla m_k(0)\|_{\mathcal{Z}} \min \left\{ \Delta_k, \frac{\|\nabla m_k(0)\|_{\mathcal{Z}}}{\beta_k} \right\}, \quad (4.2)$$

where $\kappa_0 > 0$ and $\beta_k = 1 + \sup_{s \in \mathcal{B}_k} \|\nabla^2 m_k(s)\|_{\mathcal{L}(\mathcal{Z}, \mathcal{Z}^*)}$. For all k the model m_k must approximate the true objective function $s \mapsto J(z_k + s)$ so that the true and approximate gradients at $s = 0$ satisfy

$$\|\nabla m_k(0) - \nabla J(z_k)\|_{\mathcal{Z}} \leq \xi \min\{\|\nabla m_k(0)\|_{\mathcal{Z}}, \Delta_k\}. \quad (4.3)$$

Here, $\xi > 0$ is independent of k . This condition is due to [?].

The step s_k is accepted, i.e., *successful*, if it produces sufficient decrease in the objective function J . That is, if the ratio between actual and predicted reduction,

$$\text{ared}_k = J(z_k) - J(z_k + s_k) \quad \text{and} \quad (4.4)$$

$$\text{pred}_k = m_k(0) - m_k(s_k), \quad (4.5)$$

respectively, is larger than some $\eta_1 \in (0, 1)$, $\text{ared}_k / \text{pred}_k \geq \eta_1$. Often the evaluation of J is impossible, but an approximation J_k can be computed. The subscript k indicates that the objective function approximation may change from iteration to iteration. With this approximation, we can compute the reduction

$$\text{cred}_k = J_k(z_k) - J_k(z_k + s_k). \quad (4.6)$$

To ensure convergence of the trust-region algorithm we must ensure that $|\text{ared}_k - \text{cred}_k|$ is sufficient small. The authors of [?, Sec. 10.6] require that

$$\max \{|J(z_k) - J_k(z_k)|, |J(z_k + s_k) - J_k(z_k + s_k)|\} \leq \tilde{\eta} \text{pred}_k \quad \forall k, \quad (4.7)$$

for some $\tilde{\eta} \leq \frac{1}{2}\eta_1$. This condition is similar to the condition used in [?]. The difficulty is that the constant $\tilde{\eta}$ is tied to the parameter η_1 in the trust-region algorithm used to decide the acceptance of the step. However, if error estimators used to bound the error between the exact and computed objective functions depend on unknown constants it is impossible to guarantee (4.7). We build on [?] to remove this difficulty.

We assume that there exists an estimator $\theta_k = \theta(z_k, s_k)$ for the error in the objective function so that for a constant $K > 0$,

$$|\text{ared}_k - \text{cred}_k| \leq K\theta_k \quad \forall k. \quad (4.8a)$$

Note that we need the existence of a constant K with (4.8a), but we do not need to know its value. We can reduce $\theta_k = \theta(z_k, s_k)$ below the given tolerance through, e.g., mesh or quadrature refinement. For fixed $\omega \in (0, 1)$, we control the error estimator θ_k via the following bound,

$$\theta_k^\omega \leq \eta \min \{\text{pred}_k, r_k\}, \quad (4.8b)$$

where

$$\eta < \min \{\eta_1, 1 - \eta_2\} \quad \text{and} \quad \{r_k\}_{k=1}^\infty \subset [0, \infty) \text{ satisfies } \lim_{k \rightarrow \infty} r_k = 0. \quad (4.8c)$$

Here, $\eta_2 \in (\eta_1, 1)$ is the threshold used to determine whether the trust-region radius should be increased. That is, if

$$\varrho_k \stackrel{\text{def}}{=} \frac{\text{cred}_k}{\text{pred}_k} \geq \eta_2,$$

the trust-region radius is increased. Condition (4.8b) is similar to the condition presented in [?, Sec. 10.6] with the addition of the forcing sequence r_k . Notice that, since $r_k \rightarrow 0$ as $k \rightarrow \infty$, we have that $\theta_k \rightarrow 0$ as $k \rightarrow \infty$. Therefore, for k sufficiently large, $\theta_k \leq K^{-1/(1-\omega)}$ and

$$|\text{ared}_k - \text{cred}_k| \leq K\theta_k = K\theta_k^\omega \theta_k^{1-\omega} \leq \theta_k^\omega \leq \eta \min \{\text{pred}_k, r_k\}. \quad (4.9)$$

The addition of the forcing sequence can also be used in the conditions from [?] or [?, Sec. 10.6]. In [?, p. 16], where models J_k and m_k are obtained via adaptive mesh refinement, it is assumed that if the mesh is refined infinitely often, i.e., if for a subsequence $\{k_i\}$ a new model J_{k_i} is computed via mesh refinement, then $\theta_{k_i} \rightarrow 0$ as $k_i \rightarrow \infty$. With this assumption, the explicit forcing parameter r_k in (4.8b) is not needed. Note also that $\text{pred}_k \rightarrow 0$ if $s_k \rightarrow 0$. Hence as the iteration converges, $\text{pred}_k \rightarrow 0$, and in this case one can replace $\min \{\text{pred}_k, r_k\}$ with pred_k .

The classic trust-region algorithm is listed given as follows.

ALGORITHM 4.1.

1. **Initialization:** Given $z_k, \Delta_k, 0 < \gamma_1 \leq \gamma_2 < 1, \Delta_{\max} > 0$, and $0 < \eta_1 < \eta_2 < 1$.
2. **Model Selection:** Choose a model m_k which satisfies (4.3).
3. **Step Computation:** Compute an approximate solution s_k of (4.1) that satisfies the fraction of Cauchy decrease condition (4.2).
4. **Objective Function Update:** Determine an objective function approximation J_k such that the corresponding error estimate θ_k satisfies (4.8).
5. **Step Acceptance:** Compute $\varrho_k = \text{cred}_k / \text{pred}_k$.
if $\varrho_k \geq \eta_1$ **then** $z_{k+1} = z_k + s_k$ **else** $z_{k+1} = z_k$ **end if**
6. **Trust-Region Update:**
if $z_{k+1} = z_k$ **then** $\Delta_{k+1} \in (0, \gamma_1 \|s_k\|_{\mathcal{Z}}]$
else Update Δ_{k+1} by
if $\varrho_k \leq \eta_1$ **then** $\Delta_{k+1} \in (0, \gamma_2 \|s_k\|_{\mathcal{Z}}]$ **end if**
if $\rho_k \in (\eta_1, \eta_2)$ **then** $\Delta_{k+1} \in [\gamma_2 \|s_k\|_{\mathcal{Z}}, \Delta_k]$ **end if**
if $\varrho_k \geq \eta_2$ **then** $\Delta_{k+1} \in [\Delta_k, \Delta_{\max}]$ **end if**

REMARK 4.2. For completeness we study the retrospective trust-region method of Bastin et al. [?], which we also analyzed in [?]. The retrospective trust-region determines whether to accept a step based on the current local model m_k , but when the step is accepted the retrospective algorithm updates the trust-region radius based on the new model $m_{k+1}(s) \approx J(z_{k+1} + s)$. Therefore the retrospective update may increase the trust-region radius faster if the new model m_{k+1} is better than the old one. For the retrospective trust-region algorithm, the inexact gradient condition (4.3) is replaced in [?] by

$$\|\nabla m_k(0) - \nabla J(z_k)\|_{\mathcal{Z}} \leq \xi \min\{\|\nabla m_k(0)\|_{\mathcal{Z}}, \Delta_{k-1}\}. \quad (4.10)$$

Moreover, Step 2 (Model Selection) in Algorithm 4.1 is performed following Step 5 (Step Acceptance) to produce the new model $m_{k+1}(s) \approx J(z_{k+1} + s)$ and Step 6 in Algorithm 4.1 (Trust-Region Update) is modified as follows:

- if** $z_{k+1} = z_k$ **then** $\Delta_{k+1} \in (0, \gamma_1 \|s_k\|_{\mathcal{Z}}]$
else Compute $\tilde{\varrho}_{k+1} = \text{cred}_k / (m_{k+1}(-s_k) - m_{k+1}(0))$ and update Δ_{k+1} by
if $\tilde{\varrho}_{k+1} \leq \eta_1$ **then** $\Delta_{k+1} \in (0, \gamma_2 \|s_k\|_{\mathcal{Z}}]$ **end if**
if $\tilde{\rho}_{k+1} \in (\eta_1, \eta_2)$ **then** $\Delta_{k+1} \in [\gamma_2 \|s_k\|_{\mathcal{Z}}, \Delta_k]$ **end if**
if $\tilde{\varrho}_{k+1} \geq \eta_2$ **then** $\Delta_{k+1} \in [\Delta_k, \Delta_{\max}]$ **end if**.

To prove convergence of the classic Trust-Region Algorithm 4.1 we need the following assumptions on the objective function, its approximation, and the model.

ASSUMPTIONS 4.3.

1. $J : \mathcal{Z} \rightarrow \mathbb{R}$ is twice continuously Fréchet differentiable and bounded below.
2. $J_k : \mathcal{Z} \rightarrow \mathbb{R}$ is bounded below for all k .
3. $m_k : \mathcal{Z} \rightarrow \mathbb{R}$ is twice continuously Fréchet differentiable for all k .
4. There exist $\kappa_1, \kappa_2 > 0$ such that for all $z \in \mathcal{Z}$ and for all k

$$\|\nabla^2 J(z)\|_{\mathcal{L}(\mathcal{Z}, \mathcal{Z}^*)} \leq \kappa_1 \quad \text{and} \quad \|\nabla^2 m_k(z)\|_{\mathcal{L}(\mathcal{Z}, \mathcal{Z}^*)} \leq \kappa_2.$$

Under Assumptions 4.3, one can prove the first-order convergence of Algorithm 4.1. This is a slight generalization of the convergence results in [?, ?, ?]. We prove this result in Appendix A.

THEOREM 4.4. *If Assumptions 4.3 hold, then the iterates $\{z_k\}$ generated by the classic trust-region algorithm, Algorithm 4.1, or by the retrospective trust-region algorithm, Algorithm 4.1 with Remark 4.2, satisfy*

$$\liminf_{k \rightarrow \infty} \|\nabla m_k(0)\|_{\mathcal{Z}} = \liminf_{k \rightarrow \infty} \|\nabla J(z_k)\|_{\mathcal{Z}} = 0.$$

4.2. The Gradient Condition and Adaptive Sparse Grids. To satisfy (4.3) we define the local model using the sparse-grid approximation

$$m_k(s) \stackrel{\text{def}}{=} J_{\mathcal{I}_k}(z_k + s) = \frac{1}{2} \mathbb{E}_{\mathcal{I}_k} \left[\|Cu(z_k + s) - \bar{w}\|_{\mathcal{W}}^2 \right] + \frac{\alpha}{2} \|z_k + s\|_{\mathcal{Z}}^2,$$

where $z_k \in \mathcal{Z}$ and $\mathcal{I}_k \subset \mathbb{N}_+^M$ is an admissible index set. The specific forms of $\nabla J(z)$ and $\nabla J_{\mathcal{I}_k}(z)$ give rise to the gradient error

$$\|\nabla J(z) - \nabla J_{\mathcal{I}_k}(z)\|_{\mathcal{Z}} = \left\| \sum_{\mathbf{i} \notin \mathcal{I}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\mathbf{B}^* p] \right\|_{\mathcal{Z}}.$$

We determine the admissible index set \mathcal{I}_k using the dimension-adaptive approach presented in [?]. This approach builds the admissible index set $\mathcal{I}_k = \mathcal{A}_k \cup \mathcal{O}_k$ decomposed into a set of ‘old’ indices \mathcal{O}_k and a set of active indices \mathcal{A}_k and uses integral contributions for indices in \mathcal{A}_k to estimate the error. Specifically, we use the contributions from the active set \mathcal{A}_k as a heuristic estimate for $\left\| \sum_{\mathbf{i} \notin \mathcal{I}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\mathbf{B}^* p] \right\|_{\mathcal{Z}}$. That is, we compute an index set $\mathcal{I}_k = \mathcal{O}_k \cup \mathcal{A}_k$ using Algorithm 4.5 below such that

$$\begin{aligned} & \left\| \sum_{\mathbf{i} \in \mathcal{A}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\mathbf{B}^* p] \right\|_{\mathcal{Z}} \\ & \leq \xi \min \left\{ \left\| \alpha z + \sum_{\mathbf{i} \in \mathcal{I}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\mathbf{B}^* p] \right\|_{\mathcal{Z}}, \Delta_k \right\}. \end{aligned} \quad (4.11)$$

Although there is no proof that we can bound the left-hand side with a constant times the right-hand side, our numerical examples suggest that this is typically satisfied.

The model selection and gradient computation algorithm is listed in Algorithm 4.5.

ALGORITHM 4.5. *Set $\mathbf{i} = (1, \dots, 1)$, $\mathcal{A} = \{\mathbf{i}\}$, $\mathcal{O} = \emptyset$, $r_{\mathbf{i}} = (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\mathbf{B}^* p]$ and $\beta = \beta_{\mathbf{i}} = \|r_{\mathbf{i}}\|_{\mathcal{Z}}$, $g = \alpha z + r_{\mathbf{i}}$, and $\text{TOL} = \xi \min\{\|g\|_{\mathcal{Z}}, \Delta_k\}$*

while $\beta > \text{TOL}$ **do**

Select $\mathbf{i} \in \mathcal{A}$ corresponding to the largest $\eta_{\mathbf{i}}$

Set $\mathcal{A} \leftarrow \mathcal{A} \setminus \{\mathbf{i}\}$ and $\mathcal{O} \leftarrow \mathcal{O} \cup \{\mathbf{i}\}$

Update the error indicator $\beta \leftarrow \beta - \beta_{\mathbf{i}}$

for $k=1, \dots, M$ **do**

Set $\mathbf{j} = \mathbf{i} + \mathbf{e}_k$

if $\mathcal{O} \cup \{\mathbf{j}\}$ *is admissible* **then**

Set $\mathcal{A} \leftarrow \mathcal{A} \cup \{\mathbf{j}\}$

Set $r_{\mathbf{j}} = (\Delta_1^{j_1} \otimes \cdots \otimes \Delta_M^{j_M}) [\mathbf{B}^ p]$*

Set $\beta_{\mathbf{j}} = \|r_{\mathbf{j}}\|_{\mathcal{Z}}$

Update the gradient approximation $g \leftarrow g + r_{\mathbf{j}}$

Update the error indicator $\beta \leftarrow \beta + \beta_{\mathbf{j}}$

Update the stopping tolerance $\text{TOL} = \xi \min\{\|g\|_{\mathcal{Z}}, \Delta_k\}$
end if
end for
end while
Set $\mathcal{I}_k = \mathcal{A} \cup \mathcal{O}$, $m_k(s) = J_{\mathcal{I}_k}(z_k + s)$, and $\nabla m_k(0) = g$.

If the retrospective trust-region method is used, the tolerance in Algorithm 4.5 is set to $\text{TOL} = \xi \min\{\|g\|_{\mathcal{Z}}, \Delta_{k-1}\}$; see (4.10).

4.3. Inexact Objective Functions and Adaptive Sparse Grids. To compute cred_k , we define the varying-fidelity objective function approximations

$$J_k(z) \stackrel{\text{def}}{=} J_{\mathcal{J}_k}(z) = \frac{1}{2} \mathbb{E}_{\mathcal{J}_k} [\|\mathbf{C}u(z) - \bar{w}\|_{\mathcal{W}}^2] + \frac{\alpha}{2} \|z\|_{\mathcal{Z}}^2,$$

where $z_k \in \mathcal{Z}$ and $\mathcal{J}_k \subset \mathbb{N}_+^M$ is an admissible index set.

The specific form of the objective function error for a given index set \mathcal{J}_k is

$$\begin{aligned} J(z_k) - J_k(z_k) &= \frac{1}{2} \sum_{\mathbf{i} \notin \mathcal{J}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\|\mathbf{C}u(z_k) - \bar{w}\|_{\mathcal{W}}^2] \\ J(z_k + s_k) - J_k(z_k + s_k) &= \frac{1}{2} \sum_{\mathbf{i} \notin \mathcal{J}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\|\mathbf{C}u(z_k + s_k) - \bar{w}\|_{\mathcal{W}}^2] \end{aligned}$$

and the inexact objective function evaluation condition, (4.8a), is

$$\begin{aligned} \frac{1}{2} \left| \sum_{\mathbf{i} \notin \mathcal{J}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\|\mathbf{C}u(z_k + s_k) - \bar{w}\|_{\mathcal{W}}^2 - \|\mathbf{C}u(z_k) - \bar{w}\|_{\mathcal{W}}^2] \right|^\omega \\ \leq \eta \min\{\text{pred}_k, r_k\}. \end{aligned} \quad (4.12)$$

Again, we use the dimension-adaptive approach presented in [?] to construct the admissible index set $\mathcal{J}_k = \mathcal{A}_k \cup \mathcal{O}_k$, which is decomposed into a set of ‘old’ indices \mathcal{O}_k and a set of active indices \mathcal{A}_k , and we use the contributions from the active set \mathcal{A}_k as a heuristic error estimator $\theta_k = \theta_k(z_k, s_k)$:

$$\begin{aligned} \theta_k &\stackrel{\text{def}}{=} \left| \sum_{\mathbf{i} \in \mathcal{A}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\|\mathbf{C}u(z_k + s_k) - \bar{w}\|_{\mathcal{W}}^2 - \|\mathbf{C}u(z_k) - \bar{w}\|_{\mathcal{W}}^2] \right| \\ &\approx \left| \sum_{\mathbf{i} \notin \mathcal{J}_k} (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\|\mathbf{C}u(z_k + s_k) - \bar{w}\|_{\mathcal{W}}^2 - \|\mathbf{C}u(z_k) - \bar{w}\|_{\mathcal{W}}^2] \right|. \end{aligned}$$

In addition, note that if there exists $K > 0$ such that (4.8a) holds, then (4.9) is satisfied for sufficiently large k , and global convergence is ensured. We have not verified this result for the examples in Section 5.

The objective approximation J_k in Step 4 of Algorithm 4.1 is chosen as follows.

ALGORITHM 4.6. Set $\mathbf{i} = (1, \dots, 1)$, $\mathcal{A} = \{\mathbf{i}\}$, $\mathcal{O} = \emptyset$,
 $\vartheta = \theta_{\mathbf{i}} = (\Delta_1^{i_1} \otimes \cdots \otimes \Delta_M^{i_M}) [\|\mathbf{C}u(z_k + s_k) - \bar{w}\|_{\mathcal{W}}^2 - \|\mathbf{C}u(z_k) - \bar{w}\|_{\mathcal{W}}^2]$, $c = \theta_{\mathbf{i}}$, and
 $\text{TOL} = (\eta \min\{\text{pred}_k, r_k\})^{1/\omega}$
while $|\vartheta| > \text{TOL}$ **do**

```

Select  $\mathbf{i} \in \mathcal{A}$  corresponding to the largest  $|r_{\mathbf{i}}|$ 
Set  $\mathcal{A} \leftarrow \mathcal{A} \setminus \{\mathbf{i}\}$  and  $\mathcal{O} \leftarrow \mathcal{O} \cup \{\mathbf{i}\}$ 
Update the error indicator  $\vartheta \leftarrow \vartheta - \theta_{\mathbf{i}}$ 
for  $k=1, \dots, M$  do
  Set  $\mathbf{j} = \mathbf{i} + \mathbf{e}_k$ 
  if  $\mathcal{O} \cup \{\mathbf{j}\}$  is admissible then
    Set  $\mathcal{A} \leftarrow \mathcal{A} \cup \{\mathbf{j}\}$ 
    Set  $\theta_{\mathbf{j}} = (\Delta_1^{j_1} \otimes \dots \otimes \Delta_M^{j_M})[\|\mathbf{C}u(z_k + s_k) - \bar{w}\|_{\mathcal{W}}^2 - \|\mathbf{C}u(z_k) - \bar{w}\|_{\mathcal{W}}^2]$ 
    Update the computed reduction  $c \leftarrow c + \theta_{\mathbf{i}}$ 
    Update the error indicator  $\theta_k \leftarrow \theta_k + \theta_{\mathbf{i}}$ 
  end if
end for
end while
Set  $\mathcal{J}_k = \mathcal{A} \cup \mathcal{O}$  and  $\text{cred}_k = c$ .

```

5. Numerical Results. In this section, we present the results of our fully adaptive framework for PDE-constrained optimization under uncertainty applied to two numerical examples similar to those used in [?]: optimal control of the steady 1D Burgers' equation and optimal control of the 2D Helmholtz equation. For their solution we use the classic trust-region algorithm, Algorithm 4.1, with truncated conjugate gradients to solve the trust-region subproblem [?]. Algorithmic parameters in (4.3)-(4.9) are set to $\zeta = 0.1$, $\xi = 0.01$, $\omega = 0.75$, $r_k = 0.9^k$, $\eta_1 = 0.05$, $\eta_2 = 0.75$, $\gamma_1 = 0.5$ and $\gamma_2 = 2.5$.

Implementation. As in [?], our software is built on Trilinos [?, ?]. For spatial finite element discretizations and to efficiently manage the adaptive stochastic discretizations we use the Intrepid [?] package. For additional details, see [?].

Computational Infrastructure. In contrast with [?], where we used the RedSky computing cluster at Sandia National Laboratories to perform the 2D Helmholtz numerical experiments, the Helmholtz optimal control example in Section 5.2 is solved on a single workstation with dual six-core Xeon X5680 3.33GHz processors and 24GB of RAM. The significant reduction in the required computing power is due entirely to the algorithmic advances presented in this paper. These advances pave the way for an efficient solution of optimization problems governed by 3D and time-dependent PDE models with random state variables.

5.1. Optimal Control of Steady Burgers' Equation. Our first example is the optimal control of the steady viscous Burgers' equation. The problem formulation is identical to [?]. In this paper, we change the spatial discretization slightly. We partition the domain $D = (0, 1)$ into three subdomains $D \cup \partial D = [0, 0.2] \cup [0.2, 0.8] \cup [0.8, 1]$. We mesh $[0, 0.2]$ with 80 uniform intervals; $[0.2, 0.8]$ with 16 uniform intervals; and $[0.8, 1]$ with 160 uniform intervals. Our maximum sparse-grid discretization uses level-8 isotropic Smolyak sparse grids built on 1D Clenshaw-Curtis knots. We solve the discretized nonlinear system at each sparse-grid point using Newton's method globalized with a backtracking line search.

We compare three optimization algorithms: Newton-CG on a fixed sparse grid, the trust-region method with gradient adaptivity as proposed in [?], and the fully adaptive approach developed in this paper. The computational cost of these optimization algorithms is proportional to the number of PDE solves required. Our implementation

stores the state and adjoint variables for the current and previous iterations. This avoids recomputations during the adaptivity and the truncated CG iterations.

Table 5.1 contains the comparison of the methods. As in [?], we note that the gradient-only adaptive approach (‘Grad. Adapt.’) drastically reduces the number of adjoint and sensitivity (linear) PDE solves required to obtain a minimizer — we observe a reduction by a factor of 143.9. However, the principal bottleneck of this approach is the evaluation of the high-fidelity objective function at each outer trust-region iteration, resulting in a similar number of state (nonlinear) PDE solves as in Newton-CG. On the other hand, our fully adaptive approach (‘Full Adapt.’) significantly reduces the number of state (nonlinear) PDE solves while requiring the same number of linear PDE solves as the gradient-only adaptive approach. The fully adaptive approach reduces the number of nonlinear solves by a factor of 75.0 when compared with the gradient adaptive approach and Newton-CG. Moreover, we do not sacrifice accuracy in the computed optimal controls by incorporating the additional layer of objective function inexactness. The relative error between the optimal controls computed using Newton-CG and the fully adaptive approach is 2.89×10^{-6} .

Algorithm	NonlinPDE	CP _{obj}	LinearPDE	CP _{grad}	Rel. Err.
Newton-CG	45,224 (1.0)	7,537	489,906 (1.0)	7,537	—
Grad. Adapt.	45,531 (1.0)	7,537	3,405 (143.9)	249	2.89×10^{-6}
Full Adapt.	603 (75.0)	23	3,405 (143.9)	249	2.89×10^{-6}

Table 5.1: The total number of nonlinear PDE solves (NonlinPDE), the final number of collocation points used for the objective function (CP_{obj}), the total number of linear PDE solves (LinearPDE), the final number of collocation points used for the model of the subproblem (CP_{grad}), and relative error between the controls computed using adaptivity and the control computed using Newton-CG. The numbers in parentheses in the NonlinPDE and LinearPDE columns are the ratios of the PDE counts required for Newton-CG and the PDE counts required for the adaptive approaches.

5.2. Optimal Control of Stochastic Helmholtz Equation. The second example is motivated by direct field acoustic testing [?], where the goal is to accurately shape sound pressure fields in a region of interest by using high-powered loudspeakers.

Similar to [?], we consider an example in two spatial dimensions, where the domain is $D = (-5, 5)^2$. The goal is to match the wave pressure u to a desired wave pressure $\bar{w} \in L^2(D; \mathbb{C})$ in the disk $D_R \subset D$, $D_R := \{x \in D : \|x\|_2 \leq 2\}$. In contrast to [?], where we used acoustic controls distributed in an annulus, for enhanced model fidelity we apply square-shaped acoustic controls at 20 discrete locations surrounding D_R , see Figure 5.1. The square-shaped controls are of width 0.3. The centers of the control squares are distributed equidistantly on the circle of radius 2.65 centered at the origin, with the first square center at (2.65, 0). The sides of the squares closest to D_R are tangent to the circle of radius 2.5 centered at the origin. We denote the spatial support of the control squares by D_C , i.e., the union of blue regions in Figure 5.1, left pane. To model loudspeaker enclosures, each acoustic control is surrounded on the three sides facing away from D_R by a region of higher wave speed. The desired complex-valued wave pressure is given by the plane-wave expression

$$\bar{w}(x) = \exp\left(i((k \cos \theta)x_1 + (k \sin \theta)x_2)\right)$$

where the angle of wave propagation is $\theta = \pi/4$, and $k = 10$. We call k the wave number. The real part of the desired wave pressure is shown in Figure 5.1, right pane.

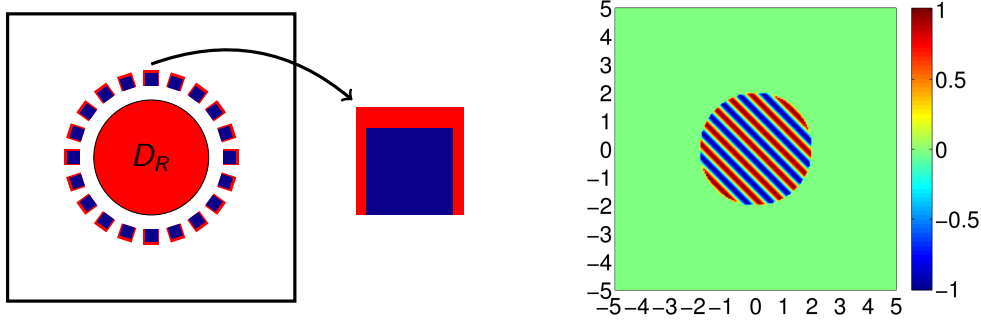


Fig. 5.1: *Left pane*: A sketch of the computational domain with the region of interest D_R and 20 loudspeakers spaced uniformly along a circle surrounding D_R . The red (lighter) color denotes uncertainty in the refraction index inside D_R and uncertainty in the loudspeaker enclosures (including thickness and sound speed). The blue (darker) color stands for the loudspeaker control regions, whose union is denoted by D_C . *Right pane*: The real component of the desired state, \bar{w} .

The optimal control problem is given by

$$\min_{z \in L^2(D; \mathbb{C})} \frac{1}{2} \int_{\Gamma} \rho(y) \int_{D_R} (u(z; y, x) - \bar{w}(x)) \overline{(u(z; y, x) - \bar{w}(x))} dx dy + \frac{\alpha}{2} \int_{D_c} z(x) \overline{z(x)} dx, \quad (5.1)$$

where $u(z; y, \cdot) = u(y, \cdot) \in H^1(D; \mathbb{C})$ for all $y \in \Gamma \equiv \Gamma_{\text{KL}} \times \Gamma_{\text{SW}} \times \Gamma_{\text{SM}}$ solves

$$-\Delta u(y, x) - K(y, x)u(y, x) = z(x) \quad \forall (y, x) \in \Gamma \times D \quad (5.2)$$

with Robin boundary conditions $\frac{\partial u}{\partial n}(y, x) = iku(y, x)$, $(y, x) \in \Gamma \times \partial D$.

In (5.2), u is the wave pressure. The random quantity $K(y, x) > 0$ and the sets Γ_{KL} , Γ_{SW} and Γ_{SM} will be specified below. The Robin boundary conditions where k is the given wave number, are a first-order approximation of the Sommerfeld radiation condition.

We study three sources of uncertainty. First, we assume that the refractive index of the medium in the region of interest D_R is random, and use the stochastic Helmholtz equation to model the governing physics, as derived, for example, in [?, ?, ?]. Specifically, in D_R we set $K(y, x) = k^2(1 + \sigma\epsilon(y, x))^2$, where the stochastic refractive index $1 + \sigma\epsilon(y, x)$ satisfies $\sigma = 0.1$, $\mathbb{E}[\epsilon(\cdot, x)] = 0$, and $\mathbb{E}[\epsilon(\cdot, x)\epsilon(\cdot, \zeta)] = C(\|x - \zeta\|)$. As in [?] we choose C to be an instance of the Matérn covariance functions,

$$C(r) = C_{\nu}(r) := \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{2\sqrt{\nu}r}{\ell} \right)^{\nu} \mathbf{K}_{\nu} \left(\frac{2\sqrt{\nu}r}{\ell} \right),$$

with parameters $\nu = \frac{11}{2}$ and $\ell = 2$. Here, $\Gamma(\nu)$ is the gamma function, and \mathbf{K}_{ν} is the modified Bessel function of the third kind [?]. We approximate the refractive index using a truncated Karhunen-Loève (KL) expansion of $\epsilon(y, x)$,

$$\epsilon(y, x) \approx \sum_{m=1}^M \epsilon_m(x) y_m,$$

with coefficients $\epsilon_m : D \rightarrow \mathbb{R}$, for $m = 1, \dots, M$, and uncorrelated random variables y_m uniformly distributed on the interval $[-\sqrt{3}, \sqrt{3}]$. This gives $\Gamma_{\text{KL}} \equiv [-\sqrt{3}, \sqrt{3}]^M$. Second, we assume that the thickness of each loudspeaker enclosure is a uniformly distributed random variable, resulting in 20 random thicknesses. Specifically, the width of each square that is the union of the blue and red regions in Figure 5.1 is drawn from $[0.3, 0.33]$, in other words $\Gamma_{\text{SW}} \equiv [0.3, 0.33]^{20}$. Third, we assume that the speed of sound in each loudspeaker enclosure is a uniformly distributed random variable. In particular, we choose the wave numbers in the red enclosure regions from $\Gamma_{\text{SM}} \equiv [0.8k_r, 1.2k_r]^{20}$, where the nominal value k_r is set to 2.287^1 . The squares of the wave numbers define the quantity $K(y, x)$ in the red enclosure regions. Outside of the enclosure regions and the region D_R we set $K(y, x) = k^2$, where $k = 10$. In summary, we consider 40 random loudspeaker parameters (including enclosure thicknesses and wave numbers) and M random region-of-interest parameters.

For the spatial discretization we use continuous Q1 finite elements on a uniform mesh of 200×200 quadrilaterals. The collocation discretizations for the optimization problem are built on 1D Clenshaw-Curtis interpolation knots. The control penalty parameter is $\alpha = 10^{-4}$. We terminate the algorithm when the norm of the model gradient falls below 10^{-6} . The overall stochastic dimension of the problem, which we denote by \dim , is given by $\dim = 40 + M$, where M is the order of the KL expansion of the refractive index. Here we consider problems with $\dim \in \{42, 44, 46, 48, 50, 60, 70, 80\}$.

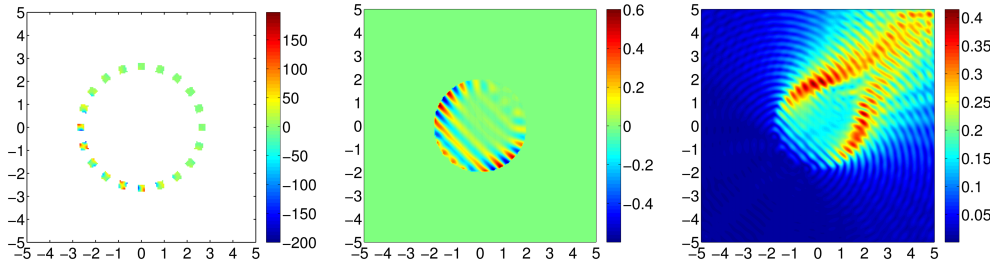


Fig. 5.2: *Left pane:* The real parts of the computed optimal controls. *Center pane:* The real part of the expected value of the optimal state, restricted to the region of interest. *Right pane:* The real part of the standard deviation of the optimal state.

Figure 5.2 shows the real parts of the computed optimal controls, the real part of the expected value of the optimal state restricted to the region of interest, and the real part of the standard deviation of the optimal state, for the largest stochastic dimension, $\dim = 80$. The results appear very similar to those obtained in [?].

Table 5.2 gives the computational cost of our algorithm as the stochastic dimension increases from $\dim = 42$ to $\dim = 80$. First, we emphasize that even for the largest dimension, $\dim = 80$, the problem can be solved using the modest computational resources described at the beginning of this section. Second, we recall that the 40-dimensional problem from [?] required 1,804,001 collocation points for the evaluation of the high-fidelity objective function, while the 80-dimensional problem here requires only 311 points due to the adaptive objective function evaluations. Third, a close examination of the stochastic dimensions explored by our algorithm reveals

¹This value is motivated by the material properties of wood-based composites. It is derived from an air-speed to wood-speed ratio of 343/1500 and the wave number in air of $k = 10$, which was used previously to define the desired pressure \bar{w} .

dim	PDE Solves	CP_{obj}	CP_{grad}	Obj. Value
42	11,543	145	145	5.2542
44	32,739	233	481	5.2637
46	60,617	243	1,453	5.2641
48	79,221	247	2,961	5.2641
50	90,157	251	4,569	5.2641
60	100,911	271	7,621	5.2641
70	103,979	291	8,233	5.2641
80	105,607	311	8,253	5.2641

Table 5.2: Computational cost of the fully adaptive trust-region algorithm applied to the Helmholtz control example. Here dim is the stochastic dimension, PDE Solves is the total number of forward and adjoint PDE solves, CP_{obj} is the final number of collocation points used by the adaptive objective function scheme, CP_{grad} is the final number of collocation points used by the adaptive subproblem (gradient) model, and Obj. Value is the computed value of the objective function at termination.

that the 40 random loudspeaker parameters are largely insignificant. The random material parameters in the region of interest account for most of the adaptive collocation performed by the algorithm, which is the reason why we report results for $dim > 40$. Finally, we note that while the number of PDE solves increases with the stochastic dimension, the increase between dimensions $dim = 60$ and $dim = 80$ is very small — only about 5%. Here the behavior of our algorithm is nearly independent of the stochastic dimension. In particular, the algorithm automatically zooms in on approximately 10 (out of 80) stochastic dimensions that are relevant to achieving objective function and gradient consistency conditions, thereby vastly reducing the effective problem size.

6. Conclusions. We have introduced a new trust-region algorithm with adaptive sparse-grid collocation for the numerical solution of optimization problems governed by PDEs with uncertain coefficients. The algorithm extends the use of consistency conditions, studied in the context of gradient computations in [?], to objective function evaluations, eliminating the need for expensive high-fidelity sparse-grid discretizations. Our fully adaptive approach builds two separate sparse grids: one to model the derivative information that is used to define the trust-region subproblem, and one to approximate the objective function. These adapted sparse grids typically contain significantly fewer points than high-fidelity grids.

The algorithm is applied to two numerical examples. The results show a vast reduction in the number of PDE solves required to compute an accurate solution of the optimization problem, when compared with the method in [?]. Moreover, the algorithm rapidly identifies the stochastic variables that are relevant to obtaining an accurate optimal solution. When the number of such variables is independent of the dimension of the stochastic space, the algorithm exhibits near dimension-independent behavior.

REFERENCES

Appendix A. Convergence Proof. We first prove results concerning step acceptance and trust-region radius update. We then use these results to prove the

first-order convergence of Algorithm 4.1. Most results presented here follow the standard proof for the classic trust-region algorithm provided in [?, Th. 4.10], although care must be taken to handle the retrospective trust-region update and the inexact objective function evaluations. To simplify the presentation, we recall the definitions of the actual, predicted and computed reductions, respectively,

$$\text{ared}_k = J(z_k) - J(z_k + s_k), \quad \text{pred}_k = m_k(0) - m_k(s_k), \quad \text{cred}_k = J_k(z_k) - J_k(z_k + s_k).$$

LEMMA A.1. *If the inexact objective function condition (4.8) hold, then for k sufficiently large*

$$\varrho_k^* \stackrel{\text{def}}{=} \frac{\text{ared}_k}{\text{pred}_k} \in [\varrho_k - \eta, \varrho_k + \eta].$$

Proof. For k sufficiently large, condition (4.8) implies

$$\varrho_k^* = \varrho_k + (\varrho_k^* - \varrho_k) \geq \varrho_k - \frac{|\text{ared}_k - \text{cred}_k|}{\text{pred}_k} \geq \varrho_k - \eta.$$

Similarly, for sufficiently large k ,

$$\varrho_k^* = \varrho_k + (\varrho_k^* - \varrho_k) \leq \varrho_k + \frac{|\text{ared}_k - \text{cred}_k|}{\text{pred}_k} \leq \varrho_k + \eta.$$

□

Lemma A.1 is used to prove the sequence of trust-region radii converges to zero.

LEMMA A.2. *Let Assumptions 4.3 and inexact objective function condition (4.8) hold. If there exists $\epsilon > 0$ such that $\|\nabla m_k(0)\|_{\mathcal{Z}} \geq \epsilon$ for k sufficiently large, then the sequence of trust-region radii, $\{\Delta_k\}$, produced by Algorithm 4.1 satisfies*

$$\sum_{k=1}^{\infty} \Delta_k < \infty.$$

Proof. First note that this lemma holds if there are only finitely many successful iterations since for sufficiently large k , $\Delta_{k+1} \leq \gamma_1 \Delta_k$.

Now, suppose there is an infinite sequence of successful iterations $\{k_i\}$. For k_i sufficiently large, Assumptions 4.3, Lemma A.1, Step 5 (Step Acceptance) of Algorithm 4.1, and the fraction of Cauchy decrease condition (4.2) imply

$$\text{ared}_{k_i} \geq \text{cred}_{k_i} - \eta \text{pred}_{k_i} \geq (\eta_1 - \eta) \text{pred}_{k_i} \geq (\eta_1 - \eta) \kappa_0 \epsilon \min \left\{ \frac{\epsilon}{1 + \kappa_2}, \Delta_{k_i} \right\}.$$

Since J is bounded below by assumption, summing the actual decrease in the objective function J gives

$$(\eta_1 - \eta) \kappa_0 \epsilon \sum_{i=1}^{\infty} \min \left\{ \frac{\epsilon}{1 + \kappa_2}, \Delta_{k_i} \right\} \leq \sum_{i=1}^{\infty} \text{ared}_{k_i} = J(z_{k_0}) - \lim_{i \rightarrow \infty} J(z_{k_i}) < \infty.$$

Therefore, $\text{ared}_{k_i} \rightarrow 0$ as $i \rightarrow \infty$ and, since $\eta_1 - \eta > 0$, we have $\sum_{i=1}^{\infty} \Delta_{k_i} < \infty$.

For every unsuccessful iteration $k \notin \{k_i\}$ the trust-region radius satisfies $\Delta_k \leq \gamma_1^{k-k_j} \Delta_{k_j}$ where $k_j \in \{k_i\}$ is the largest index of a successful iteration such that $k_j < k$. The convergence of geometric series and the above result imply that

$$\sum_{k \notin \{k_i\}} \Delta_k \leq \frac{1}{1-\gamma_1} \sum_{i=1}^{\infty} \Delta_{k_i} \quad \text{and} \quad \sum_{k=1}^{\infty} \Delta_k \leq \left(1 + \frac{1}{1-\gamma_1}\right) \sum_{i=1}^{\infty} \Delta_{k_i} < \infty.$$

This proves the desired result. \square

Lemma A.2 is used to obtain a contradiction. To arrive at this contradiction, we first prove that Algorithm 4.1 produces a successful step for sufficiently large k .

LEMMA A.3. *Let Assumptions 4.3 hold. If there exists $\epsilon > 0$ such that $\|\nabla m_k(0)\|_{\mathcal{Z}} \geq \epsilon$, then in Algorithm 4.1 the condition $\varrho_k \geq \eta_2 > \eta_1$ is satisfied for k sufficiently large.*

Proof. By Taylor's theorem, there exist $t_1, t_2 \in [0, 1]$ such that

$$\begin{aligned} \text{ared}_k &= \langle \nabla J(z_k), s_k \rangle_{\mathcal{Z}} + \frac{1}{2} \langle \nabla^2 J(z_k + t_1 s_k) s_k, s_k \rangle_{\mathcal{Z}} \\ \text{pred}_k &= \langle \nabla m_k(0), s_k \rangle_{\mathcal{Z}} + \frac{1}{2} \langle \nabla^2 m_k(t_2 s_k) s_k, s_k \rangle_{\mathcal{Z}}. \end{aligned}$$

The above Taylor expansions and Assumptions 4.3 imply

$$|\text{ared}_k - \text{pred}_k| \leq \xi \Delta_k^2 + \frac{1}{2} (\kappa_1 + \kappa_2) \Delta_k^2.$$

Furthermore, the fraction of Cauchy decrease condition (4.2), Lemma A.2, Assumptions 4.3, and $\|\nabla m_k(0)\|_{\mathcal{Z}} \geq \epsilon$ imply that for sufficiently large k ,

$$\text{pred}_k \geq \kappa_0 \|\nabla m_k(0)\|_{\mathcal{Z}} \min \left\{ \Delta_k, \frac{\|\nabla m_k(0)\|_{\mathcal{Z}}}{1 + \kappa_2} \right\} \geq \kappa_0 \epsilon \min \left\{ \Delta_k, \frac{\epsilon}{1 + \kappa_2} \right\} \geq \kappa_0 \epsilon \Delta_k.$$

Combining these inequalities gives

$$|\varrho_k^* - 1| \leq \frac{\xi \Delta_k + \frac{1}{2} (\kappa_1 + \kappa_2) \Delta_k}{\kappa_0 \epsilon}$$

for sufficiently large k . Therefore, by Lemma A.2, $\varrho_k^* \rightarrow 1$ as $k \rightarrow \infty$. By Lemma A.1, $\varrho_k^* + \eta \geq \varrho_k \geq \varrho_k^* - \eta$ and thus $\lim_{k \rightarrow \infty} \varrho_k \in [1 - \eta, 1 + \eta]$. Since $\eta < 1 - \eta_2$, there exists some k such that $\varrho_k \geq \eta_2$. \square

Now we are able to prove the convergence Theorem 4.4 for the classic trust-region algorithm, Algorithm 4.1.

Proof. (of Theorem 4.4 for the classic trust-region algorithm) For contradiction, suppose there exists $\epsilon > 0$ such that $\|\nabla m_k(0)\|_{\mathcal{Z}} \geq \epsilon$. By Lemma A.2, $\lim_{k \rightarrow \infty} \Delta_k = 0$. However, by Lemma A.3, $\varrho_k \geq \eta_2$ for all k sufficiently large. Since $\varrho_k \geq \eta_2$ implies $\Delta_{k+1} \geq \Delta_k$, this contradicts $\lim_{k \rightarrow \infty} \Delta_k = 0$. Thus, $\liminf_{k \rightarrow \infty} \|\nabla m_k(0)\|_{\mathcal{Z}} = 0$.

By the inexact gradient condition (4.3), $\liminf_{k \rightarrow \infty} \|\nabla m_k(0)\|_{\mathcal{Z}} = 0$ implies $\liminf_{k \rightarrow \infty} \|\nabla f(z_k)\|_{\mathcal{Z}} = 0$. \square

To prove convergence of the retrospective trust-region algorithm, in addition to achieving a successful step, i.e., $\varrho_k \geq \eta_1$, we must show that $\tilde{\varrho}_{k+1} \geq \eta_2$.

LEMMA A.4. *Let Assumptions 4.3 hold. If there exists $\epsilon > 0$ such that $\|\nabla m_k(0)\|_{\mathcal{Z}} \geq \epsilon$ for k sufficiently large, then in the retrospective trust-region algorithm, Algorithm 4.1 with Remark 4.2, the conditions $\varrho_k \geq \eta_1$ and $\tilde{\varrho}_{k+1} \geq \eta_2$ hold for all k sufficiently large.*

Proof. Lemma A.3 proves $\varrho_k \geq \eta_1$ for all k sufficiently large. By Taylor's theorem, there exists $t_3 \in [0, 1]$ such that

$$m_{k+1}(-s_k) - m_{k+1}(0) = -\langle \nabla m_{k+1}(0), s_k \rangle_{\mathcal{Z}} + \frac{1}{2} \langle \nabla^2 m_{k+1}(t_3 s_k) s_k, s_k \rangle_{\mathcal{Z}}.$$

This equality, the expansion of pred_k in the proof of Lemma A.3, and Assumptions 4.3 imply

$$|\text{pred}_k - (m_{k+1}(-s_k) - m_{k+1}(0))| \leq \|\nabla m_{k+1}(0) - \nabla m_k(0)\|_{\mathcal{Z}} \Delta_k + \kappa_2 \Delta_k^2.$$

To bound this further, notice that

$$\begin{aligned} \|\nabla m_{k+1}(0) - \nabla m_k(0)\|_{\mathcal{Z}} &\leq \|\nabla m_{k+1}(0) - \nabla J(z_k + s_k)\|_{\mathcal{Z}} \\ &\quad + \|\nabla J(z_k + s_k) - \nabla J(z_k)\|_{\mathcal{Z}} \\ &\quad + \|\nabla J(z_k) - \nabla m_k(0)\|_{\mathcal{Z}}. \end{aligned} \quad (\text{A.1})$$

We bound the first and third quantities on the right-hand side of (A.1) using (4.10), and the second quantity using the differentiability of J , namely,

$$\|\nabla J(z_k + s_k) - \nabla J(z_k)\|_{\mathcal{Z}} = \left\| \int_0^1 \nabla^2 J(z_k + ts_k) s_k dt \right\|_{\mathcal{Z}} \leq \kappa_1 \Delta_k.$$

This proves that

$$|\text{pred}_k - (m_{k+1}(-s_k) - m_{k+1}(0))| \leq (\xi \Delta_k + \xi \Delta_{k-1} + \kappa_1 \Delta_k) \Delta_k + \kappa_2 \Delta_k^2,$$

which implies the bounds

$$\text{pred}_k - \tilde{\epsilon}_k \Delta_k \leq (m_{k+1}(-s_k) - m_{k+1}(0)) \leq \text{pred}_k + \tilde{\epsilon}_k \Delta_k$$

with $\tilde{\epsilon}_k = (\xi \Delta_k + \xi \Delta_{k-1} + \kappa_1 \Delta_k + \kappa_2 \Delta_k)$. The fraction of Cauchy decrease condition and the assumption that $\|\nabla m_k(0)\|_{\mathcal{Z}} \geq \epsilon$ imply

$$(m_{k+1}(-s_k) - m_{k+1}(0)) \geq (\kappa_0 \epsilon - \tilde{\epsilon}_k) \Delta_k. \quad (\text{A.2})$$

Since $\tilde{\epsilon}_k$ converges to zero by Lemma A.2, the right-hand side of (A.2) is positive for sufficiently large k . These bounds and Lemmas A.1 and A.3 imply that, for sufficiently large k ,

$$\begin{aligned} \tilde{\varrho}_{k+1} &= \left(\frac{\text{cred}_k}{\text{pred}_k} \right) \left(\frac{\text{pred}_k}{m_{k+1}(-s_k) - m_{k+1}(0)} \right) = \varrho_k \left(\frac{\text{pred}_k}{m_{k+1}(-s_k) - m_{k+1}(0)} \right) \\ &\geq \varrho_k \left(1 - \frac{\tilde{\epsilon}_k \Delta_k}{m_{k+1}(-s_k) - m_{k+1}(0)} \right) \geq \varrho_k \left(1 - \frac{\tilde{\epsilon}_k}{\kappa_0 \epsilon - \tilde{\epsilon}_k} \right) \geq \varrho_k - \frac{(1 + \eta) \tilde{\epsilon}_k}{\kappa_0 \epsilon - \tilde{\epsilon}_k}. \end{aligned}$$

Hence, for sufficiently large k , we have $\tilde{\varrho}_{k+1} \geq \eta_2$, as desired. \square

The first-order convergence result for the retrospective trust-region algorithm, Algorithm 4.1 with Remark 4.2, can now be proven by following the proof of Theorem 4.4 for the classic trust-region algorithm, replacing Lemma A.3 by Lemma A.4.